

AD-A056 664

NAVAL ENVIRONMENTAL PREDICTION RESEARCH FACILITY MON--ETC F/G 9/2
APPLICATION OF AN ARRAY PROCESSOR IN SATELLITE IMAGE PROCESSING--ETC(U)
JUN 77 P C ZELENY, R E NAGLE

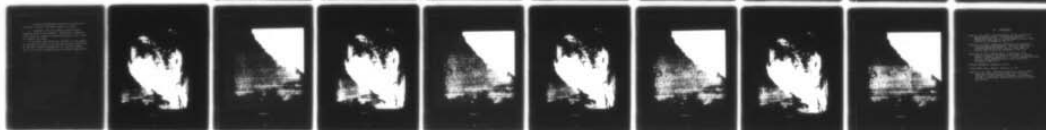
MIPR-2165-71-8010

UNCLASSIFIED

NEPRF-TM-77-01

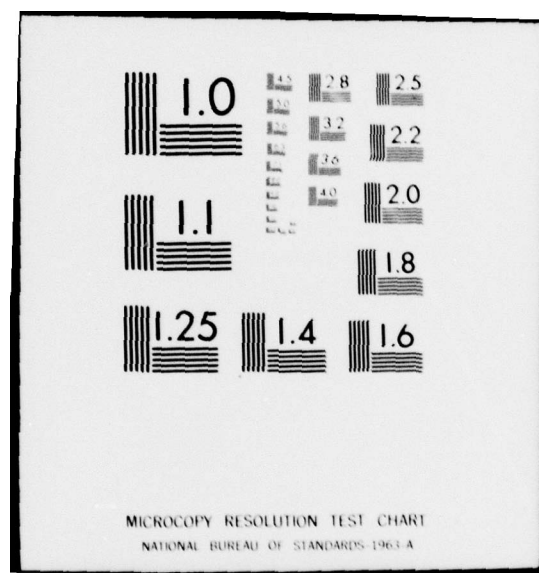
NL

| OF |
AD
A056664



END
DATE
FILMED
9-78

DDC



AD A056664



LEVEL II

NAVENVPREDRSCHFAC
Technical Memorandum
TM 77-01

10
SC

APPLICATION OF AN ARRAY PROCESSOR IN
SATELLITE IMAGE PROCESSING

by

Pimporn C. Zeleny and Roland E. Nagle

Naval Environmental Prediction Research Facility
Monterey, California 93940

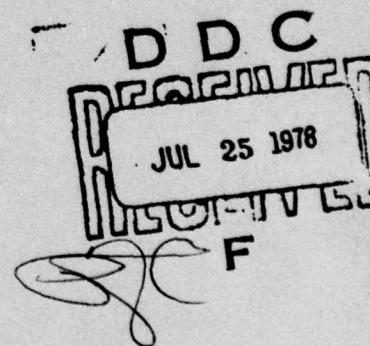
Final Report

for

Department of the Army
United States Army Electronic Command
Atmospheric Sciences Laboratory
White Sands Missile Range, NM 88002

30 June 1977

Approved for Public Release;
Distribution Unlimited



78 07 18 03 3

AD NO. _____
DDC FILE COPY

(14) NEPRF-TM 77-01
UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|-----------------------|---|
| 1. REPORT NUMBER NAVENVPREDRSCHFAC Tech. Memorandum TM 77-01 | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle) Application of an Array Processor in Satellite Image Processing. | | 5. TYPE OF REPORT & PERIOD COVERED Final Report, 1 Oct 1976-30 June 1977 |
| 6. AUTHOR(s) Pimporn C./Zeleny Roland E./Nagle | | 7. PERFORMING ORG. REPORT NUMBER |
| 8. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Environmental Prediction Research Facility Monterey, CA 93940 | | 9. CONTRACT OR GRANT NUMBER(s) MIPR-#2165-71-8010 Seg. 0036-J019043,44 |
| 10. CONTROLLING OFFICE NAME AND ADDRESS Department of the Army U.S. Army Electronics Command Atmospheric Sciences Laboratory | | 11. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS MIPR LAW-W43P6S 7TB010 |
| 12. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) U.S. Army Electronics Command Atmospheric Sciences Laboratory White Sands Missile Range, NM | | 13. REPORT DATE June 1977 |
| | | 14. NUMBER OF PAGES 20 42, 22 P. |
| | | 15. SECURITY CLASS. (of this report) UNCLASSIFIED |
| | | 16a. DECLASSIFICATION/DOWNGRADING SCHEDULE |
| 16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited. | | |
| 17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) | | |
| 18. SUPPLEMENTARY NOTES | | |
| 19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Array processor Satellite Image Processing Fast Fourier Transform | | |
| 20. ABSTRACT (Continue on reverse side if necessary and identify by block number) An array processor was integrated into an interactive graphics display system which incorporates a NOVA 2/10 mini-computer. Difficulties experienced with the installation of this hardware are described. Limitations of the software documentation are itemized. An application of the array processors FFT capability are applied to the suppression of noise in satellite imagery is presented. Results are discussed. | | |

DD FORM 1 JAN 73 1473

EDITION OF NOV 65 IS OBSOLETE
S N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

447 279

hc

TABLE OF CONTENTS

| | | |
|------|---|----|
| I. | INTRODUCTION | 2 |
| II. | BACKGROUND | 3 |
| III. | HARDWARE DESCRIPTION AND IMPLEMENTATION | 3 |
| | A. MFFT Hardware Processor Function | 4 |
| | B. Installation | 5 |
| | C. Testing | 5 |
| IV. | SOFTWARE DESCRIPTION, DOCUMENTATION AND USAGE | 6 |
| V. | APPLICATIONS | 8 |
| VI. | CONCLUSIONS AND RECOMMENDATIONS | 10 |
| VII. | REFERENCES | 20 |

| | |
|---------------------------------|---|
| ACCESS FOR | |
| NTS | Section <input checked="" type="checkbox"/> |
| DDC | Section <input type="checkbox"/> |
| NAVJAG | <input type="checkbox"/> |
| JST/ST/ST | |
| BY | |
| DISTRIBUTION/AVAILABILITY CODES | |
| Doc | or SPECIAL |
| A | |

APPLICATION OF AN ARRAY PROCESSOR IN SATELLITE IMAGE PROCESSING

I. INTRODUCTION

One of the most impressive facets of environmental satellites is the immense amount of data which they produce. Previously, this vast amount of data has relegated consideration of digital processing of these data to only the largest of centralized computer facilities. Even when large, high speed computers are applied to satellite data processing, it has not been possible to process all the data produced by one satellite system such as the Defense Meteorological Satellite Program (DMSP) on a global basis or in quasi-real-time. Those data that are processed are still resident at the centralized facility and the problem remains of disseminating the required information to battlefield commanders. A method of extracting the inherent information contained in high resolution satellite imagery has not as yet been achieved nor has a cost-effective method been devised for disseminating this information to numerous remote locations from a central facility.

The evolution of mini-, micro- and special-purpose computer technology over the past ten years now allows consideration of direct acquisition of digital satellite data at local and regional tactical sites and full digital processing for mission orientated information. This document is the final report of an exploratory study of the use of a hard-wired array processor for satellite data processing. The array processor is a peripheral device to one of the mini-computers contained in the Naval Environmental Prediction Research Facility's (NEPRF) Satellite Data Processing and Display System (SPADS).

This work has been performed as a cooperative effort between NEPRF and the Atmospheric Sciences Laboratory of the U.S. Army Electronics Command, White Sands Missile Range, NM.

II. BACKGROUND

Applications of Fast Fourier Transforms (FFT) have been widely used by engineers and scientists specializing in signal processing and analyses. Until recently, the FFT was performed by computer software. This process required relatively large computers and considerable amounts of computer time. The amount of computer resources required to perform FFT's on significant amounts of satellite data has precluded real-time application of this technique and has also limited its application to satellite data processing on a research basis to a few pilot studies. (See for example, Sikula, 1974 and Dart, 1974, 1975 and 1976).

Within the past few years, several vendors have designed and marketed various types of signal processing hardware, usually referred to as array processors. These devices are special-purpose computers that are capable of performing arithmetic on two or more sets of numbers stored in two or more arrays. The arithmetic operations are performed on the arrays in parallel which greatly increases execution speed in comparison to normal general purpose computer execution. Execution speeds several orders of magnitude faster than general purpose computers can be achieved with these devices.

III. HARDWARE DESCRIPTION AND IMPLEMENTATIONS

The array processor chosen for this study was restricted by the requirement that it must be compatible with Data General NOVA 2/10 computers currently used at SPADS. A hardware array arithmetic processor marketed by ELSYTEC, called 306/MFFT, was chosen for this particular reason.

The MFFT board was received on November 23, 1976 along with a set of paper tapes of source programs of the software package, and a user's manual. The software package will be discussed in section IV. The following paragraphs will describe the processor, its installation and testing.

A. MFFT Hardware Processor Function

As an array arithmetic processor, the MFFT is capable of performing the following functions:

1. Arithmetic (single and double precision)

MFFT performs addition, subtraction and a logical operation between each (single or double) word element in one array and the corresponding element in a second array, and returns the result in the third array.

2. Multiply (single and double precision)

3. Divide (single and double precision)

4. Square-Root (single and double precision)

5. Array accumulate

All single precision elements of an array are accumulated into a double precision register.

6. Multiply/Integrate

Same as double precision multiply but each element of the resultant array contains the running sum of the most significant half of the present product and all previous products.

7. Multiply/Accumulate

Same as a single precision multiply, except the individual products are not stored but are accumulated as a double precision value.

B. Installation

The MFFT board was advertised as using only one slot of the NOVA computer, however, the thickness of the board was such that the slot below was not available for use, and for practical purpose, the board physically occupied two slots of the CPU.

The only wiring required on the back plane of the NOVA is to connect the data channel priority line into the chain. Therefore installation of an MFFT board is very simple.

A first attempt was made to install the MFFT in the expansion chassis of the computer where there were several unused slots and also where there was an ample power supply for the board (+5 volts, +8 amps required). However, it was discovered that the MFFT board could not function in the expansion chassis. This problem was due to the time delay on the repeater bus. The time delay resulted in memory address line not getting settled in time to address the proper word in memory. The board was then moved to one of the empty slots in the main chassis. While the board was there, the computer could not function and gave an indication that the system was overloaded, even though the computation showed that there was enough current to supply the MFFT board. It was not possible to measure the actual current drawn by the board while it was operating to check if it actually drew more than 8 amps which was the amount specified by the vendor.

The final solution was to move two existing boards in the main chassis, the multiplexor board and the disk controller, to the expansion chassis so that there would be sufficient power supply for the MFFT.

C. Testing

After the CPU overloading problem had been solved, MFFT diagnostic program, which was delivered with the

hardware, was executed. The program failed. There were indications that the timing on the MFFT board was off and the memory content was not fetched or stored at proper times. After consulting with the Data General engineer, the vendor had to perform hardware modification to the MFFT board until the timing problem was resolved and the diagnostic program executed properly.

It is to be noted that the diagnostic software package provided was not adequate to detect hardware malfunction of the MFFT board; this situation still exists.

IV. SOFTWARE DESCRIPTION, DOCUMENTATION AND USAGE

The MFFT purchased price included a set of assembly language programs to allow basic operation of the board. These programs are:

1. A hardware diagnostic program called MTBF.
2. An FFT test program called MTA4
3. A set of subroutines for use with MTBF and MTA4 which can also be used by a programmer to utilize various functions of the board. These subroutines are:
 - a. Cosine table calculation
 - b. FFT subroutine
 - c. Real valued array conversion
 - d. Magnitude of complex spectrum
 - e. Complex and conjugate multiply
 - f. Hanning subroutine
 - g. Processor driving subroutine
 - h. Read processor magnitude
 - i. Calculate PSI word
 - j. Convert magnitude code to flag code
 - k. Cosine subroutine

All programs were delivered in "source" form punched on paper tapes. Each program was read into the NOVA, assembled, and then stored in program libraries as instructed by the user's manual. Documentation available was a user's manual which has a user's guide as one of its sections.

Two difficulties arose while attempting to use the software package:

a. The numerical format used internally by the MFFT is neither a FORTRAN standard integer format nor floating point format. In order to use the MFFT, a user must perform his own conversion of all numbers (float or fixed) into MFFT format. Additionally, scaling factors for those numbers must be kept current at all times.

b. A user must write all his application programs in an assembly language in order to use the subroutines provided by the vendor.

A further discussion with the vendor revealed that there was another set of software available, at an additional cost, to provide an interface between a FORTRAN program and the MFFT subroutines. The MFFT FORTRAN interface programs consist of 15 subroutines callable from a FORTRAN program. Three of these subroutines perform the scaling and conversion of an array of FORTRAN floating point numbers into MFFT fixed point format and vice versa. However, there was no provision for converting FORTRAN integers into MFFT fixed format; this is essential when working with digital satellite data. The availability of the FORTRAN interface programs simplified the use of MFFT to a certain extent. However, a large amount of time was spent in determining the input/output parameters in the calling sequences because the documentation was not clear. For example, many key parameters in the array computation needed to be calculated and prestored before

array processing was performed. It was not possible to extract this information from the FORTRAN documentation. On occasion, it was necessary to study the assembly listing to determine the definition of the FORTRAN.

There were no examples on how the MFFT subroutines are interrelated and in which order should they be executed. In other words, a lack of good documentation created more work than necessary. Experimentation with the programs and verbal communication with the vendor's engineer solved part of the problems.

V. APPLICATIONS

There are many applications to which an array processor can be used in satellite data processing. However, noise is an inherent problem to all satellite data processing and the extraction of environmental information in a high noise situation was chosen for evaluation of the MFFT board.

The data consisted of a NOAA direct readout Automatic Picture Transmission (APT) pass acquired at NEPRF and digitized on the Direct Readout Data Converter (DRDC). The infrared data was processed to enhance sea surface temperature features potentially existing in a cloud-free area off the California coast. A section of this image (128 pixels by 64 lines) is shown in Figure 1 and again in Figure 2 as a 2 times zoomed image. As can be observed, there are obvious variations in grey-shades (temperatures) but the definition of specific features are masked by "noise" of two types: A line by line variation in the DC component and high frequency variations along each line. In an attempt to eliminate these effects, the following procedure was accomplished:

1. For each image line in the section (128 pixels):
 - a. Calculated an average value of samples.

b. Eliminated DC components by subtracting the average value from the original sample values.

c. Performed a forward FFT on the modified samples.

2. After all 64 lines have been processed,

a. performed a forward FFT on the averages of 64 lines,

b. eliminate the upper half of the spectral values (33rd to 64th), then performed an inverse transform on the modified data. The result is the new 64 entries of average values.

3. For each line, performed an inverse FFT and use the corresponding new average entry in reconstructing the pixel values.

4. The new section of the image was inserted in the original image as shown in Figure 3 and as a 2 times zoomed image in Figure 4.

5. A histogram of the IR data within the section was generated. Four IR value intervals were chosen and each interval was replaced by a single value to enhance the contrast. The result is shown in Figure 5 and as a 2 times zoomed image in Figure 6.

The choice of replacing IR values were arbitrary. The value chosen here were such that the processed section can be seen with the original background. Figures 7 and 8 showed a different set of replacement values chosen to enhance the contrast. However, in this case, the remaining part of the image faded out.

In order to test the accuracy of the MFFT, we performed a forward FFT on 64 complex points, then performed an inverse FFT on the result of the forward FFT. The final result was

accurate within about $\pm 1\%$ for 8 bit data. Two of the possible causes of this problem are:

1. Numerical accuracy was lost due to signal conversion for different numerical modes, i.e.,

Integer \longleftrightarrow Floating point \longleftrightarrow MFFT fixed

2. Loss of accuracy was due to the MFFT itself because it can only use 15 bits in fractional form.

A software FFT program was executed in the same experiment in order to compare the result and the efficiency of the software and hardware FFT. Higher accuracy was obtained from the software program. However, for the analysis of 64×64 pixels, the time required by the software FFT was about 30 minutes (real-time), while the program using the hardware FFT only required 56 seconds (real-time). This is about a factor of 30 to 1 improvement on processing time between the hardware and the software FFT.

VI. CONCLUSIONS AND RECOMMENDATIONS

It has been demonstrated that a hardware FFT can be very useful in satellite data processing. This experiment has shown that, solving the same problem, hardware FFT is about 30 times faster than software FFT. The amount of CPU time saved would be available for other processing functions in the mini-computer. As for the main memory usage, MFFT does not have its own storage area and therefore requires the use of the main memory in the same manner as the software FFT.

As far as getting the software to work with hardware FFT is concerned, it is recommended that the following items should be studied very carefully:

1. A software package should be available at a level that an application user is not required to have the hardware knowledge in order to use it.

2. Software documentation should be available at a high level of detail with many examples of usages.

3. Diagnostic and test programs should perform extensive tests on the hardware. Additionally, numerical tests should be capable of accepting external input as well as internal test input.

If the above requirements are not met, a large amount of time would be spent on learning the hardware, generation of the basic software package and its interface before a user will be able to benefit from the hardware availability.



Figure 1

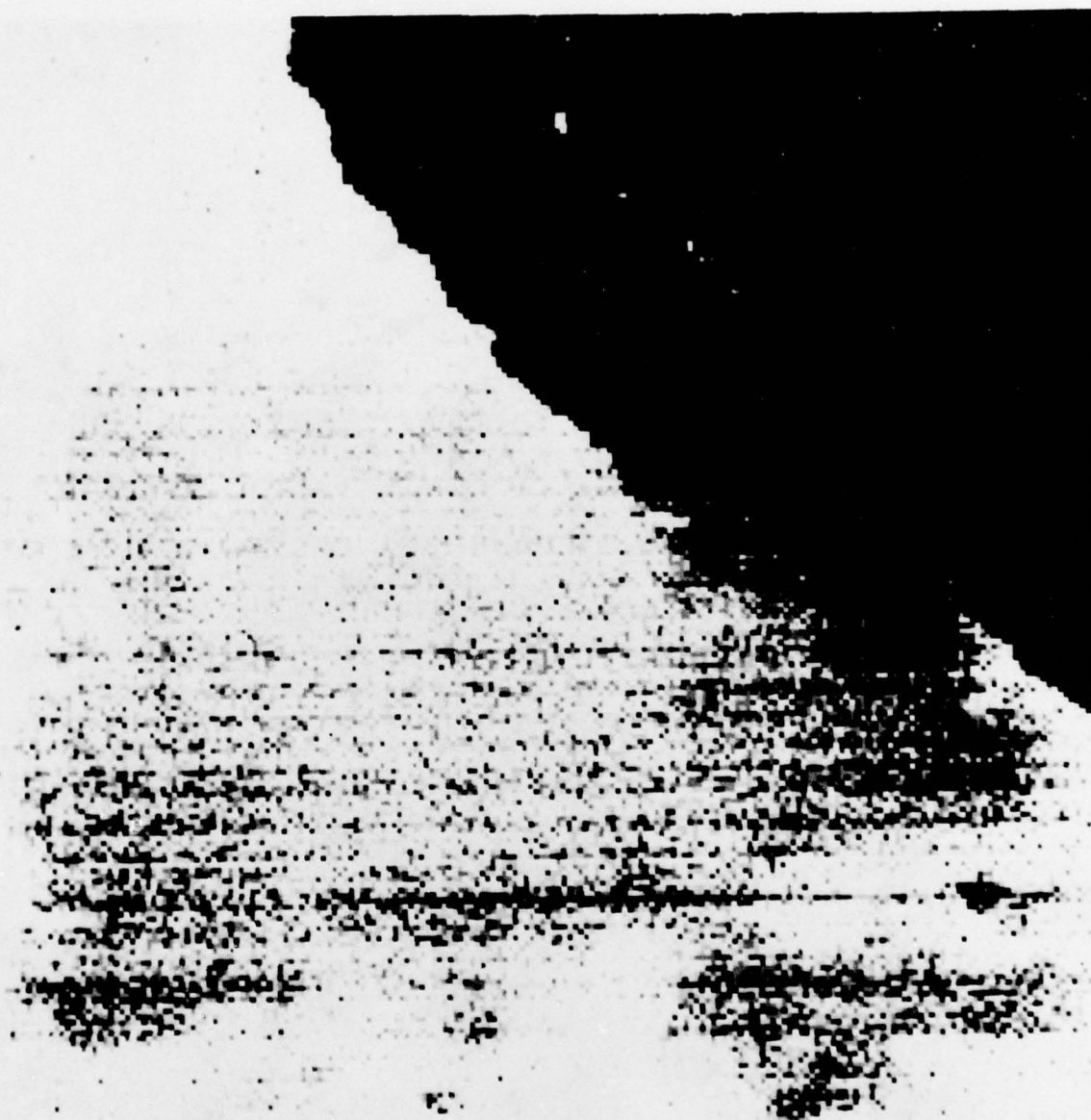


Figure 2



Figure 3



Figure 4



Figure 5



Figure 6



Figure 7

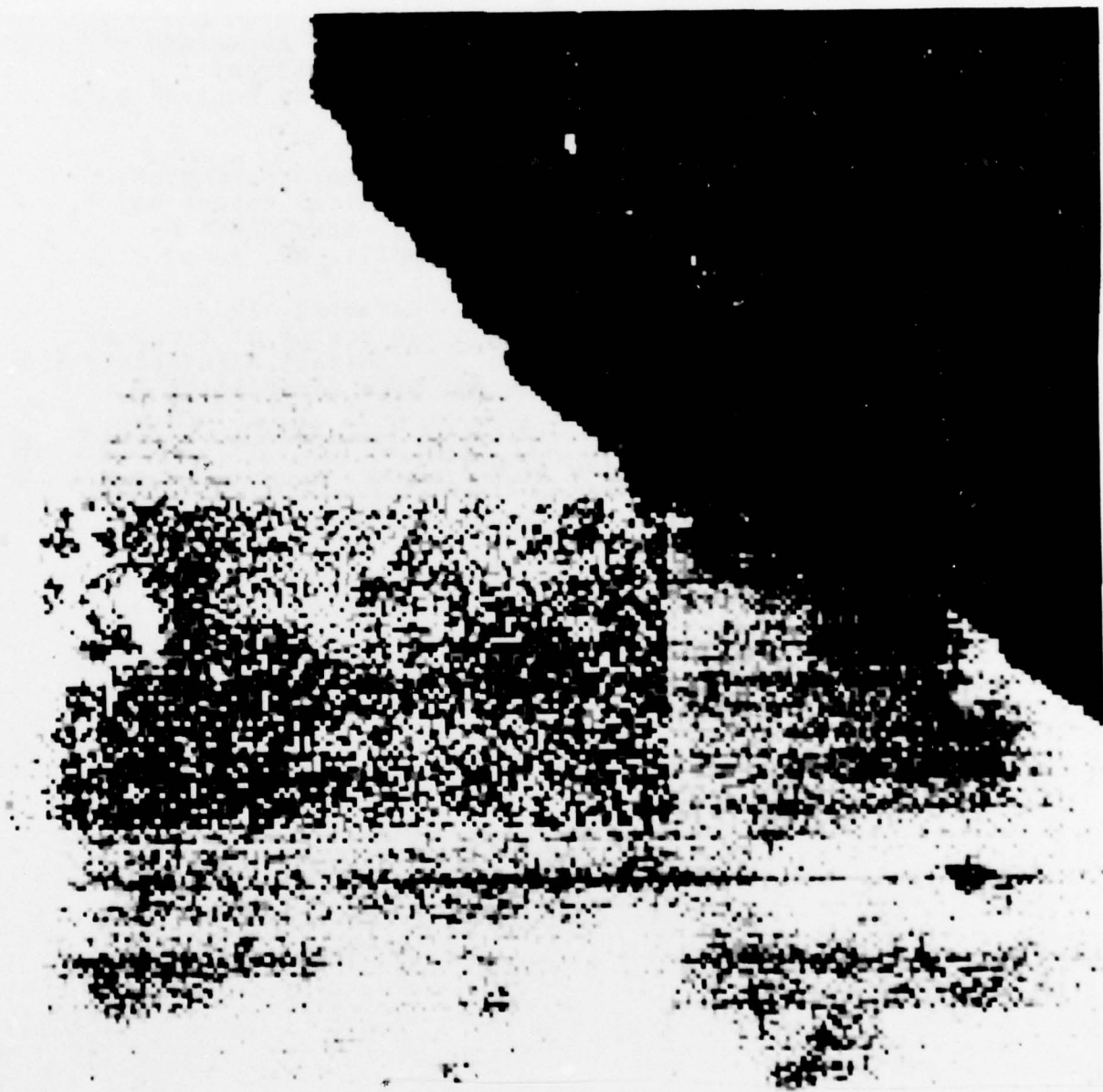


Figure 8

VII. REFERENCES

- Dartt, D. G., 1976: A pilot analysis of the curvature of extratropical storms. Final Report, Contract N00228-75-C-2363, for U.S. Navy, NEPRF by Control Data Corporation, Minneapolis, MN, 21 p.
- Dartt, D. G., 1975: Objective tropical storm parameters from satellite cloud patterns. Technical Report No. 1, Contract N00228-75-C-2363, for U.S. Navy NEPRF by Control Data Corporation, Minneapolis, MN, 23 p.
- Dartt, D. G., M. S. Ulstad and R. A. Weinberg, 1974: Objective methods for locating the center of tropical storms. Technical Report No. 1, Contract N66856-3029-5500, for U.S. Navy, NEPRF by Control Data Corporation, Minneapolis, MN, 16 p.
- ELSYTEC FORTRAN IV Interface to MFFT.
- ELSYTEC MFFT User's Manual, February 7, 1975.
- Sikula, G. J., 1974: Spectral Signature of several cloud types and information extraction from very high resolution visual satellite radiances - preliminary results. Proceedings, American Meteorological Society, pp. 478-492.